(2)

AD-A161 075

DEVELOPING TECHNICAL DOCUMENTATION STANDARDS FOR NOHIMS

W. M. PUGH
D. M. HAMILTON

REPORT NO. 85-1

NAVAL HEALTH RESEARCH CENTER
P.O. BOX 85122
SAN DIEGO, CALIFORNIA 92138-9174

NAVAL MEDICAL RESEARCH AND DEVELOPMENT COMMAND
BETHESDA, MARYLAND

DEVELOPING TECHNICAL DOCUMENTATION STANDARDS FOR NOHIMS

William M. Pugh and D. M. Hamilton
Environmental Medicine Department
Naval Health Research Center
P. O. Box 85122
San Diego, California 92138-9174

To expedite communication of our research, this is a preprint of a paper submitted to MUMPS USERS' GROUP Fourteenth Annual Meeting and will be published in their 1985 Proceedings.

Report No. 85-1 was supported by the Naval Medical Research and Development Command, Department of the Navy, under research Work Unit M0933-PN.003-0001. The views presented in this paper are those of the authors. No endorsement by the Department of the Navy has been given or should be inferred.

SUMMARY

Previously, Naval technical documentation standards have been developed using a static, top-down, structured approach that is oriented toward languages such as COBOL and FORTRAN. Because the Navy Occupational Health Information Management System (NOHIMS) uses the Massachusetts General Hospital Utility Multi-Programming System (MUMPS) language, which is dynamic in nature, it was felt that new technical documentation standards had to be developed to accomodate its flexibility. This paper describes a transactional approach to documentating the NOHIMS system. Examples of documentation worksheets, as well as a diagram depicting the levels of documentation and their relationship to the transaction center, are provided. This method of documentation may be advantageous for others who are working with dynamic or evolving systems.

BACKGROUND

At the Naval Health Research Center (NHRC), Point Loma, California, the Navy Occupational Health Information Management System (NOHIMS) is being developed for the Navy's Occupational Health and Safety Program.[1,2] Because it is being developed as a research project, it is a dynamic system with goals that change as the system evolves. To accomodate the dynamic nature of the project, MUMPS was selected as the programming language for the NOHIMS effort. MUMPS' unique data management system was designed to be used in the development of complex, interactive application systems.[3] In contrast to other high level languages, MUMPS allows the application designer flexibility when handling data with imprecise specifications.[4]

Although NOHIMS has been running successfully at the Naval Air Rework Facility, Naval Air Station, North Island, the continued operation and maintenance of NOHIMS requires that the system be documented. Current Navy documemtation standards [5] were reviewed and appeared to be oriented toward languages such as Cobol and Fortran.

The Navy technical documentation standards are defined using four documents; Data Base Specifications, System or Subsystem Specifications, Program Specifications, and a Maintenance Manual. This standard assumes that the system will be designed using a top-down approach. Program Specifications describe detailed, well structured programs that call single task modules to perform system's functions or subfunctions, outlined in the System or Subsystem Specifications. Data base requirements and data flows are completey defined prior to beginning the programming of the system. The Maintenance Manual is described as an update or amplification of the Program Specification. Using the top-down approach to describe a system, and documenting it as such, works well within a closed system. However, this approach is oriented toward a fixed goal and, as such, does not take into consideration the following factors:

1. The system may have multiple goals.
2. New data elements may be needed at a later time.
3. The set of operations to complete various procedures may undergo re-definition.
4. In a dynamic system, it is nearly impossible to predict how the data will be used in the future.

Therefore, MUMPS provided the software environment that allowed the dynamic nature of NOHIMS to be met, but a new documentation approach was needed. The purpose of this paper is to describe the approach taken at NHRC to develop the documentation needed to operate and maintain NOHIMS. It is believed that the approach used is general and is applicable for other MUMPS-based systems.

APPROACH

The approach taken to develop documentation standards for NOHIMS was a transactional approach. According to Yourdon and Constantine,[6] transaction analysis is used to describe complex systems in which a variety of different options can be selected from a central point. As a rule this central point, or transaction center, occupies the upper level of the system. Transactions are initiated by calling lower level routines or modules to initiate a transaction. However, in large, complex systems, the transaction center may be at a lower level than the function it calls. There may also be transaction centers at lower levels, called actic levels, and these are referred to as action centers.

A transaction or action center is the point where transactions or processes are initiated and ultimately return. A center is found by locating the point at which data is transformed as a function of a module by splitting input data into several discrete output streams.

By using this strategy, the documentation specifications emphasize the data flows rather than the actual coding or program structure. Thus, the technical documentation can be condensed into one document, the Maintenance Manual, which consists of the Module Documentation, Test Documentation, and Special Notes. This document is a technical tool for programmers and analysts who are responsible for ongoing software support after the system is operational. Its purpose is to aid analysts and programmers when they troubleshoot problems or when they need to revise or modify the system.

Therefore, the documentation required for NOHIMS consists of user, operator, and technical documentation. The user and operator documentation provides non-technical personnel with information regarding the objectives of a system, the functions it performs, and how to use it. Thus, the user and operator documentation is not unique for MUMPS. However, the technical documentation is unique because it accomodates MUMPS' flexibility.

TECHNICAL DOCUMENTATION

Technical Documentation standards developed for NOHIMS is a transactional approach to documentation. A system may be designed which consists of only two levels; a main driver and a transaction center. However, most systems have sufficient complexity to warrant intermediate levels of documentation. In the documentation standard developed for NOHIMS, four levels have been defined. These levels are processor, transaction, action, and routine. The relationship among these levels is shown in figure 1.

The Processor level is the first level of a system. At the processor level, the primary system functions are identified. Transactions are received and dispatched to the appropriate transaction level for processing.

The Transaction level is the second level of the system where all subfunctions of the system are defined in terms of processes or operations. If a module performs many subfunctions or complex operations, it should be separated into two or more action level modules.

The Action level describes the aggregate of operations, procedures, or routines that occur within each transaction. When the transaction is lengthy or complicated, the transaction is separated into smaller components called actions.

The Routine level is the lowest level in the system. It is the actual coding level and is responsible for carrying out the details of the actions required to complete the transaction. One of the purposes of creating small, loosely coupled, internally cohesive routines is to facilitate programming design, system implementation, and subsequent maintenance or revisions. Routines should be functionally separate so that each performs a specific task or a few closely related tasks that no other routine performs.

DOCUMENTATION SPECIFICATIONS

Every level has a specific documentation requirement. The Processor Specification is a detailed technical document prepared by an analyst for the use of programmers in writing modules needed to implement a project. It describes the data to be input, the transactions to be performed, and the data to be output. It also provides important and related information such as disposition of data, security classification requirements, and the relationship of each transaction module to other modules in the same system or in other systems.

The Transaction Specification is a detailed technical document prepared by an analyst for programmers to use when writing the transaction modules. It describes the data entry and exit points. It also provides important and related information such as transformations of data, and the relationship of each transaction module to other modules in the system. Transaction specification documentation is divided into four sections which are designated the Transaction Summary, Summary of Requirements, Environmental Description, and Design Details. To assist the user in the development of the required documentation, various work sheets have been developed. Work sheets developed to assist the Transaction Summary and Summary of Requirements documentation are shown in figures 2 and 3.

The Action Specification is written by the programmer and references the transaction. It describes the data to be input, output and transformed in this module. It also provides important and related information such as the relationship of each action module to other module levels in the

system. Documentation at the Action level is broken down into four sections. These include the Action Summary, Summary of Requirements, Process Cross Reference Page, and Design Details. Again, work sheets similar to those used for the Transaction level are available.

The Routine Specification is a detailed technical document prepared by the programmer to meet the specifications defined by the analyst. It describes the function of the routine, data to be manipulated, operations to be performed, and the relationship of the routine to other module levels. Only data that is critical to the understanding to the routine need be described in detail. Routine documentation consists of the Routine Abstract, Summary of Requirements, Software Support, and Design Documentation. The work sheets provided for Routine documentation includes one for the Routine Abstract, one for the Summary of Requirements, and one for listing critical variables. The work sheet for developing Routine Abstract information is essentially the same as the one shown in figure 2 for Transaction Summary documentation. However, the work sheets shown in figures 4 and 5 are unique to the Routine Specification.

CONCLUSION

Just as MUMPS is designed to accomodate dynamic environments where goals and data elements may be changing over time, the documentation method developed for NOHIMS is designed to be flexible. Thus, this method of documentation may be advantageous for others who are working with developmental, dynamic or evolving systems.

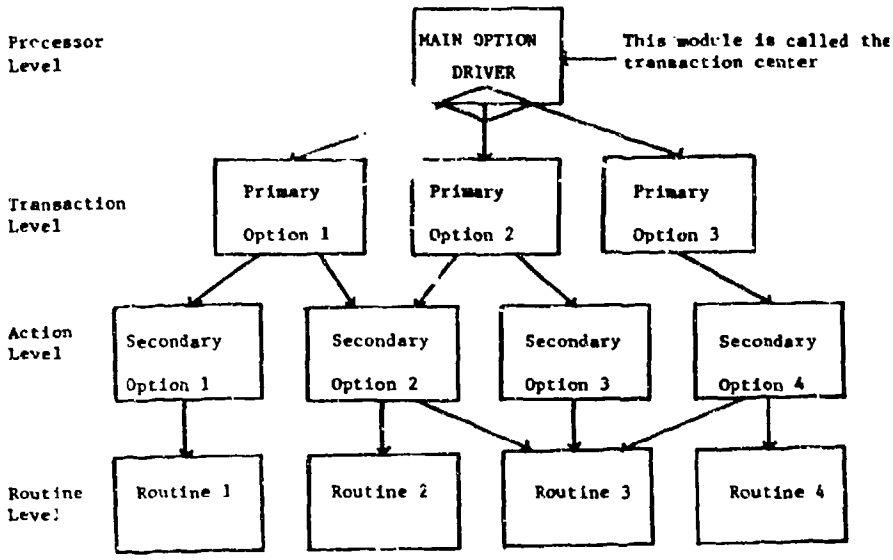

Figure 1. Levels of documentation and their relationship to the transaction centered system.

TRANSACTION SUMMARY PAGE

SYSTEM: _____ DATE: _____

SYSTEM FUNCTION: _____ PAGE: _____

SYSTEM SUBFUNCTION/TRANSACTION: _____ REV.# _____

TYPE OF TRANSACTION: (a) I/O (b) PROCESS (c) UTILITY

PURPOSE OF TRANSACTION: _____

_____

_____

_____

DISPOSITION OF OUTPUT: _____

_____

_____

INPUTS TO TRANSACTION: _____

_____

_____

RETENTION PERIOD: _____

SECURITY/PRIVACY: _____

_____

_____

NOTES: _____

_____

_____

Figure 2. Work sheet for documenting Summary information at the Transaction level.

TRANSACTION SUMMARY REQUIREMENTS

SYSTEM: _____ DATE: _____

SYSTEM FUNCTION: _____ PAGE: _____

SYSTEM SUBFUNCTION/TRANSACTION: _____

TRANSACTION DESCRIPTION: _____

_____

_____

_____

_____

FLEXIBILITY: _____

_____

_____

_____

DESCRIPTION OF ERROR CRITERIA AND RECOVERY: _____

_____

_____

_____

REPORT FORMAT REQUIREMENTS: _____

_____

_____

_____

Figure 3. Worksheet for documenting requirements at transaction level.

ROUTINE LEVEL SUMMARY OF REQUIREMENTS

SYSTEM: _____ DATE: _____

ROUTINE: _____ REV.#: _____

ROUTINE DESCRIPTION: _____

_____

_____

_____

_____

LIST OF CRITICAL VARIABLES: _____

DESCRIPTION OF ERROR CRITERIA AND RECOVERY: _____

_____

_____

_____

_____

_____

_____

NOTES: _____

_____

_____

_____

_____

_____

_____

Figure 4. Worksheet for documenting requirements at the Routine level.

LIST OF CRITICAL VARIABLES

ROUTINE: _____ PAGE ___ of ___

NAME OF VARIABLE: _____

DESCRIPTION: _____

SPECIFICATIONS: _____

_____

SOURCE OF VARIABLE: _____

NAME OF VARIABLE: _____

DESCRIPTION: _____

SPECIFICATIONS: _____

_____

SOURCE OF VARIABLE: _____

NAME OF VARIABLE: _____

DESCRIPTION: _____

SPECIFICATIONS: _____

_____

SOURCE OF VARIABLE: _____

NAME OF VARIABLE: _____

DESCRIPTION: _____

SPECIFICATIONS: _____

_____

SOURCE OF VARIABLE: _____

Figure 5. Worksheet for documenting the critical variables at the Routine level.

REFERENCES

1. Pugh, William M. & Beck, Donald D. Preliminary Specification for a Navy Occupational Health Information Monitoring System (NOHIMS). Report No. 81-36, Naval Health Research Center, San Diego, California, 1981.
2. Pugh, William M., Beck, Donald D. & Ramsey-Klee, D.M. An Overview of the Naval Occupational Health Informational Monitoring System (NOHIMS), MUG Quarterly, 129-138, Spring, 1983.
3. Rowie, Jack E. Evolution of an application tool-- A Case study of MUMPS. Proceedings of the Sixteenth Annual Hawaii International Conference on System Sciences, 2 334-342, 1983.
4. Moraff, Howard, MUMPS for medicine -- Where we are and where we're going. Proceedings of the Sixteenth Annual Hawaii International Conference on on System Sciences, 2, 332-333, 1983.
5. SECNAVINST 5233.1B, Automated Data System Documentation Standards. Washington, D.C., Department of the Navy, 1979.
6. Yourdon, E. and Constantine, L.L. Structured Design: Fundamentals of a Discipline of Computer Program and System Design. New Jersey, Prentice Hall Inc., 1979

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1 REPORT NUMBER 85-1	2. GOVT ACCESSION NO. AD-A161075	3. RECIPIENT'S CATALOG NUMBER
4 TITLE (and Subtitle) DEVELOPING TECHNICAL DOCUMENTATION STANDARDS FOR NOHIMS		5. TYPE OF REPORT & PERIOD COVERED Interim 6. PERFORMING ORG. REPORT NUMBER
7 AUTHOR(s) William M. Pugh and D.M. Hamilton		8. CONTRACT OR GRANT NUMBER(s) M0933-PN.003-0001
9 PERFORMING ORGANIZATION NAME AND ADDRESS Naval Health Research Center P. O. Box 85122 San Diego, CA 92138-9174		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11 CONTROLLING OFFICE NAME AND ADDRESS Naval Medical Research & Development Command Naval Medical Command, National Capital Region Bethesda, MD 20814		12. REPORT DATE January 1985 13. NUMBER OF PAGES 11
14 MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Commander, Naval Medical Command Department of the Navy Washington, DC 20372		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimted.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release; dis ribution unlimited.

18. SUPPLEMENTARY NOTES

To be presented at the 14th Annual Meeting of the MUMPS USERS' GROUP at Chicago, Illinois 10-14 June 1985.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

NOHIMS
MUMPS
Documentation Standards
Transaction Analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Previously, Naval technical documentation standards have been developed using a static, top-down, structured approach that is oriented toward languages such as COBOL and FORTRAN. Since NOHIMS usus the MUMPS language, which is dynamic by nature, this technical documentation had to be redesigned to accommodate its flexibility. This paper describes a transactional approach to documenting the NOHIMS system. Examples of documentation worksheets, as well as a diagram depicting the levels of documentation and their relationship to the transaction center, are provided. This method of documentation may be advantageous for

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

others who are working with dynamic or evolving systems.